Job Description Documentation

Task-Job Matrix

	Mapper	Reducer	Other
Task1			Task1Driver
Job1	Task1PlaceMapper	Task1PlaceReducer	Task1GroupComparator TextIntPair TextIntPartitioner
Job2	Task1PlaceMergeMapper Task1PhotoMergeMapper	Task1MergeReducer	TextIntPairGroupComparator TextIntPair TextIntPartitioner
Job3	Task1SortMapper	Task1SortReducer	
Task2			Task2Driver
Job1	Task2Job1Mapper	Task2Job1Reducer	IntWritableDescendingComparator
Task3			Task3Driver
Job1	Task3Job1Mapper1 Task3Job1Mapper2	Task3Job1Reducer	TextIntIPairGroupComparator TextIntIPair
Job2	Task3Job2Mapper1 Task3Job2Mapper2	Task3Job2Reducer	TextIntIPairGroupComparator TextIntPair
Job3	Task3Job3Mapper	Task3Job3Reducer	Task3Job3Combiner
Job4	Task3Job4Mapper	Task3Job4Reducer	TextIntIPairGroupComparator TextIntIPair
Job5	Task3Job5Mapper	Task3Job5Reducer	TextIntIPairGroupComparator TextIntIPair

Task1 Part

Run Command:	hadoop jar comp5349-chun.jar assign1.Task1Driver [place input] [photo input] [output]
For example	hadoop jar comp5349-chun.jar assign1.Task1Driver /share/place.txt /share/photo task1out
Estimation Duration on 10G data	1min 25sec

Job1:

Input	place-id, woeid, latitudelongitude, place-name, place-type-id, place-url
Output	locality-name, place-id

Job1 is designed to group /share/places with type of 22 together with places with type of 7, omitting other type of places.

Job2:

Input	locality-name, place-id -> [place-id,order], locality-name photo-id,owner,tags,date-taken,place-id,accuracy -> [place-id,order], “1”
Output	locality-name, numofphotos

Job2 is for joining /share/photo data and output of job1 and counting numofphotos according to same place-id. However, the numofphotos is not the real number of same locality-name, it just aggregated on place-id. So job3 is needed.

Job3:

Input	locality-name, numofphotos
Output	locality-name, numofphotos

Job3 is mainly for alphabetical sort output of job2 based on locality-name. Meanwhile, it restart a new aggregation process after job2 based on locality-name.

TextIntPair
This class is designed for key used in mapreduce which require secondary sort. Text stores real key, the Intwritable is used for order.

TextIntPairPartition
This class is used cooperating with TextIntPair, in that case the data with same real key go to the same reducer.

Task1GroupComparator
This group comparator is designed for aggregating places which share same string slice. In that case, neighbourhood level places and locality level places can group together.

Task2 Part

Run Command:	hadoop jar comp5349-chun.jar assign1.Task2Driver [task1out] [output]
For example	hadoop jar comp5349-chun.jar assign1.Task2Driver task1out task2out
Estimation Duration on 10G data	19sec

Job1:

Input	locality-name, numofphotos

Output	locality-name, numofphotos

Job1 is for sort output of task1 based on numofphotos, reducer number is set to 1 in that case top 50 results can be chosen through a counter in reducer side.

IntWritableDescendingComparator
This class is designed in aim of inversing the IntWritable default sort order

Task3 Part

Run Command:	Run Command: hadoop jar comp5349-chun.jar assign1.Task3Driver [task2out] [photo input] [output]
For example	hadoop jar comp5349-chun.jar assign1.Task3Driver task2out /share/photo task3out
Estimation Duration on 10G data	3min 11sec

Job1
Join Task1 Job1 temporary output (locality-name, place-id) and Task2 output (locality-name, numofphotos), get (locality-name, numofphotos, photo-id)

Job2
Join Task3 Job1 output (locality-name, numofphotos, photo-id) and the whole photo collection (photo-id,owner,tags,date-taken,place-id,accuracy) , get (locality-name, numofphotos, tags, date-taken)

Job3
This job take Task3 Job2 output (locality-name, numofphotos, tags, date-taken)
as input to split tags into single tag and count them when they are grouped together, the output is like (locality-name, numofphotos, tag, freq)

Job4
This job take previous job output as input, counting tag frequency based on same key, the output is like (locality-name,numberOfPhotos, [(tag1:freq1) (tag2:freq2) ... (tag10:freq10))]

Job5
This job is for sorting result from job4 according to numberOfPhotos, the output format is the same of job4

TextIntIPair
This class is the same as TextIntPair except data sorted in inverse order.

Appendix

Output on HDFS location:

Task1 output	/user/cshe6391/task1out
Task2 output	/user/cshe6391/task2out
Task3 output	/user/cshe6391/task3out